REMARKS/ARGUMENTS

Applicants amended claims 1, 17, and 33 to make corrections to informalities to overcome the Examiner's objections.

1. Claims 1-14, 17-30, and 33-46 are Patentable Over the Cited Art

The Examiner rejected claims 1-14, 17-30, and 33-46 as anticipated (35 U.S.C. §102) by DelMonaco ("IBM WebSphere Studio Asset Analyzer: Overview"), Mowen ("IMS Newsletter", Summer 2001) and Kahm "Using WebSphere Studio Asset Analyzer", (March 2004). Applicants traverse for the following reasons.

The Examiner rejected claims 5, 7, 9, 13, 21, 23, 25, 37, 39, and 41 as anticipated (35 U.S.C. §102) by both DelMonaco and Kahm. Applicants submit that the anticipation rejection of these claims under 35 U.S.C. §102 is improper because, according to the Manual of Patent Examination and Procedure (MPEP) "[a] claim is anticipated only if each and every element as set forth in the claim is found, either expressly or inherently described, in a single prior art reference." MPEP, Sec. 2131, pg. 2100-73 (8th Ed., Rev. 2004). The anticipation rejection is improper because it relies on two references, DelMonaco and Kahm, not a single reference as required. For the dependent claims, such as claims 5, 7, 21, 23, 37, and 39, the Examiner relied on DelMonaco for the base claim limitations and Kahm for the dependent claim limitations.

Moreover, Applicants traverse the application of Kahm that the Examiner located because Kahm is dated March 2004 and thus cannot be prior art to this application having an October 2001 filing date.

Accordingly, Applicants submit that the rejection of claims 5, 7, 9, 13, 21, 23, 25, 37, 39, and 41 is improper and should be withdrawn.

Claims 1, 17, and 33 concern enabling analysis of proposed changes to program statements in a source code files, and require: generating a data structure indicating a plurality of program statements in one or more source code files affected by one of a plurality of selected program statements to change in one or more source code files, wherein the data structure indicates a hierarchical relationship of the effect of program statements on one another resulting from change to the selected program statements; and processing the data structure to display information on the hierarchical relationship of the affect of the program statements on one another resulting from changes to the selected program statements.

-The Examiner cited pg. 10, pg. 3, lines 7-10, and pg. 3, lines 7-10 as disclosing the claim requirement of generating a data structure indicating a plurality of program statements in one or more source code files affected by one of a plurality of selected program statements to change in one or more source code files, wherein the data structure indicates a hierarchical relationship of the effect of program statements on one another resulting from change to the selected program statements. (Office Action, pg. 3) Applicants traverse.

The cited pg. 10 shows a GUI providing information on program statements and program files where the element is referenced or set. This cited pg. 10 nowhere discloses the claim requirement of indicating program statements affected by selected program statements to change, wherein the data structure indicates a hierarchical relationship of the effect of program statements on one another. There is now showing in the text or figure on the cited pg. 10 of indicating a hierarchical relationship of the effect of program statements on one another.

The cited pg. 3 mentions that once the inventory of the program artifacts is complete, impact analysis questions can be asked about the information in the database, such as what parts within and outside of an application need to be considered, what is affected when a block of code is changed. Nowhere in this cited pg. 3 is there any disclosure of a data structure indicating a hierarchical relationship of the effect of program statements on one another. Instead, the cited pg. 3 mentions that one may ask questions on what is affected when a block of code is changed. There is no disclosure of a data structure indicating a hierarchical relationship of the effect of program statements on one another as claimed.

The cited pg. 4 mentions that a database has metadata on application artifacts, which is checked periodically. Page 4 further mentions that artifacts in the database are organized based on their natural hierarchy below a "site" artifact. A site artifact represents a logical partition (LPAR) on a given machine, and that each artifact is associated with a site when inventoried. Although the cited pg. 4 mentions how artifacts are organized based on their site, i.e., LPAR, there is no disclosure of the claimed data structure indicating a hierarchical relationship of the effect of program statements on one another. Instead, the cited page 4 discusses how artifacts are organized based on their site/LPAR, not based on a hierarchical relationship of the effect of program statements on one another as claimed. Applicants submit organizing artifacts in a hierarchy below their "site" or LPAR is different from and does not disclose the claim

requirements of a data structure indicating a hierarchical relationship of the effect of program statements on one another.

The Examiner cited pgs. 8-11 as disclosing the claim requirement of processing the data structure to display information on the hierarchical relationship of the affect of the program statements on one another resulting from changes to the selected program statements. (Office Action, pg. 3) Applicants traverse.

The cited pg. 8 discusses searches that identify components that contain certain data elements and you can use these lists to assemble a statement of work that includes the potential impacts across data elements and outside the specific application that you are working on. The cited pgs. 9-11 discuss evaluating application code by showing a count of statements. The "Summary" mentions that after the complete inventory is taken, one can navigate through application assets, browse source files, and drill down to lower and lower levels to evaluate or implement a proposed change.

Notwithstanding that the cited DelMonaco discusses how one may access a database having information on program artifacts, nowhere do the cited pgs. 8-11 disclose the claimed display of information on the hierarchical relationship of the effect of program statements on one another. The figures shown on the cited pages do not disclose displaying the hierarchical relationship of an effect of program statements on one another.

Accordingly, claims 1, 17, and 33 are patentable over the cited art because the cited DelMonaco does not disclose all the claim requirements.

Claims 2-8, 18-24, and 34-40 are patentable over the cited art because they depend from one of claims 1, 17 and 33, which are patentable over the cited art for the reasons discussed above. Moreover, the following discussed dependent claims provide further grounds of distinction over the cited art.

Claims 2, 18, and 34 depend from claims 1, 17, and 33 and further require that generating the data structure comprises: generating an element object in the data structure for one selected program statement to change; and generating element objects in the data structure for program statements that are directly or indirectly affected by the selected program statements, wherein the element object representing one program statement is defined as a child element object to the element object for one program statement whose output parameter comprises the input parameter of the program statement represented by the child element object.

The Examiner cited pg. 10 as disclosing the claim requirement that the element object representing one program statement is defined as a child element object to the element object for one program statement whose output parameter comprises the input parameter of the program statement represented by the child element object. (Office Action, pg. 4) Applicants traverse.

The cited pg. 10 shows a screen shot of components including impacts across data elements. The cited figure shows a component CS-DEPT-CAR identified as children, a CS-PASS-AREA component identified as the parent, and also program files where the data element is referenced. Although the cited figure has a component CD-DEPT-CHAR identified as children, nowhere does this cited figure disclose that the element object representing one program statement is defined as a child element object to the element object for one program statement whose output parameter comprises the input parameter of the program statement represented by the child element object. There is no disclosure that the cited CD-DEPT-CHAR represents a program statement defined as a child element object to the element object for one program statement whose output parameter comprises the input parameter of the program statement represented by the child element object. In other words, there is not disclosure that the CS-PASS-AREA has an output parameter that comprises the input parameter of the CDS-DEPT-CHAR identified as the child, or CS-DEPT-CHAR. Instead, the cited pg. 10 mentions data elements known as children, the location of the element and programs where the data element is referenced.

Accordingly, claims 2, 18, and 24 provide additional grounds of patentability over the cited art because their additional requirements are not disclosed in the cited art.

Applicants amended claims 6, 18, and 34 to correct minor format errors.

Independent claims 9, 25, and 41 concern analyzing proposed changes to program statements in a source code file, and require: receiving a data structure indicating a plurality of program statements affected by one of a plurality of selected program statements to change, wherein the data structure indicates a hierarchical relationship of the effect of program statements on one another; displaying the hierarchical relationship of the program statements affected by one of the selected program statements; receiving user input indicating selection of one of the displayed program statements; and enabling editing of the selected program statement.

First off, Applicants note that the anticipation rejection of claims 9, 25, and 41 and claims 10-16, 26-32, and 42-48 that depend there from should be withdrawn because this anticipation

rejection does not rely on a single reference, but multiple references. Moreover, these claims are further distinguished over the cited art for the reasons discussed below.

The Examiner cited the above discussed sections of DelMonaco as teaching/disclosing the claim requirements that the data structure indicates a hierarchical relationship of the effect of program statements on one another and displaying the hierarchical relationship of the program statements affected by one of the selected program statements. Applicants submit that these requirements distinguish over DelMonaco for the reasons discussed above with respect to claims 1, 17, and 33 and the requirements of indicating and displaying a hierarchical relationship of the effect of program statements on one another.

Moreover, Applicants submit that it was improper for the Examiner to apply Kahm with respect to certain limitations because applying multiple references is not a proper anticipation rejection and because the later Kahm is not prior art to the earlier subject patent application.

Accordingly, Applicants submit that claims 9, 25, and 41 are patentable over the cited art because the cited combination does not teach, suggest or disclose all the claim requirements and because the rejection is improper.

Claims 10-16, 26-32, and 42-48 are also patentable over the cited art because they depend from claims 9, 25, and 41, which are patentable over the cited art for the reasons discussed above.

2. Claims 15, 16, 31, 32, 47, and 48 are Patentable Over the Cited Art

The Examiner rejected claims 15, 16, 31, 32, 47, and 48 as obvious (35 U.S.C. §103(a)) by DelMonaco, Mowen, Kahm, and Minard (U.S. Patent No. 6,247,020). Applicants traverse.

First off, these claims are patentable over the cited art because they depend from one of claims 9, 25, and 41.

Moreover, Applicants submit that this rejection is improper to the extent it relies on Kahm, which is not suitable prior art to the present application for the reasons discussed above.

Still further, certain of the below discussed dependent claims provide further grounds of distinction over the cited art.

Claims 15, 31, and 47 recite that the hierarchical relationship of the program statements are displayed in a first pane on a graphical user interface panel and wherein the plurality of

program statements displayed in editing mode are displayed in a second pane on the graphical user interface.

The Examiner cited FIG. 4A of Minard as teaching the additional requirements of these claims. (Office Action, pg. 10) Applicants traverse.

The cited FIG. 4A shows a browser that lets the user explore, edit, design and debug code in one window. A navigation pane 410 displays a list of documents, the content pane 450 displays the document and the structure pane 450 displays the structure of the document. (Minard, col. 8, lines 40-51, FIG. 4A) Nowhere does the cited Minard anywhere teach or suggest displaying a hierarchical relationship of how program statements effect each other in a first pane and a plurality of program statements in a second pane in editing mode. Instead, the cited Minard just shows program content in one pane and file names in the other. Nowhere is there any teaching, suggestion or mention of displaying in one pane a hierarchical relationship of how program statements effect each other and in another pane program statements to edit.

Accordingly, the additional requirements of claims 15, 31, and 47 provide further grounds of patentability over the cited art.

Conclusion

For all the above reasons, Applicant submits that the pending claims 1-48 are patentable over the art of record. Applicants have not added any claims. Nonetheless, should any additional fees be required, please charge Deposit Account No. 09-0460.

The attorney of record invites the Examiner to contact him at (310) 553-7977 if the Examiner believes such contact would advance the prosecution of the case.

Dated: March 22, 2005

By: _____
David W. Victor
Registration No. 39,867

Please direct all correspondences to:

David Victor
Konrad Raynes & Victor, LLP
315 South Beverly Drive, Ste. 210
Beverly Hills, CA 90212
Tel: 310-553-7977
Fax: 310-556-7984